# Digital Signatures with Memory-Tight Security in the Multi-Challenge Setting

Denis Diemert Kai Gellert Tibor Jager Lin Lyu

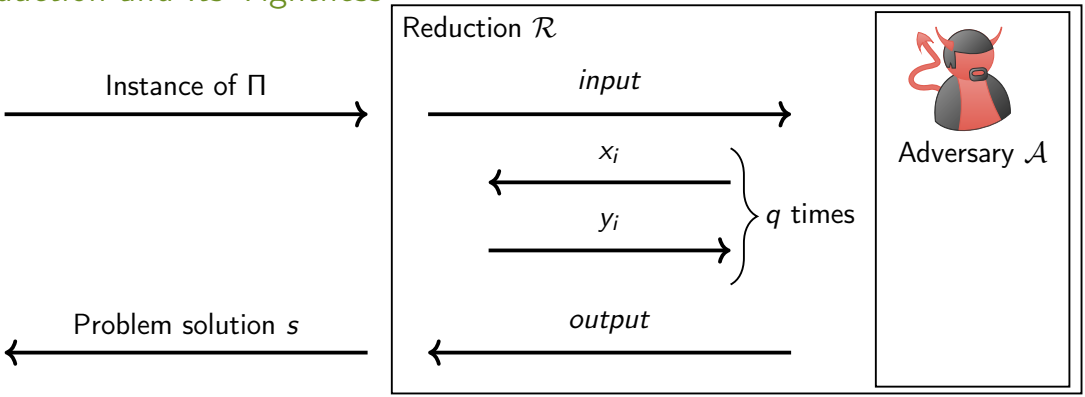
IT Security and Cryptography Group
University of Wuppertal

ASIACRYPT 2021

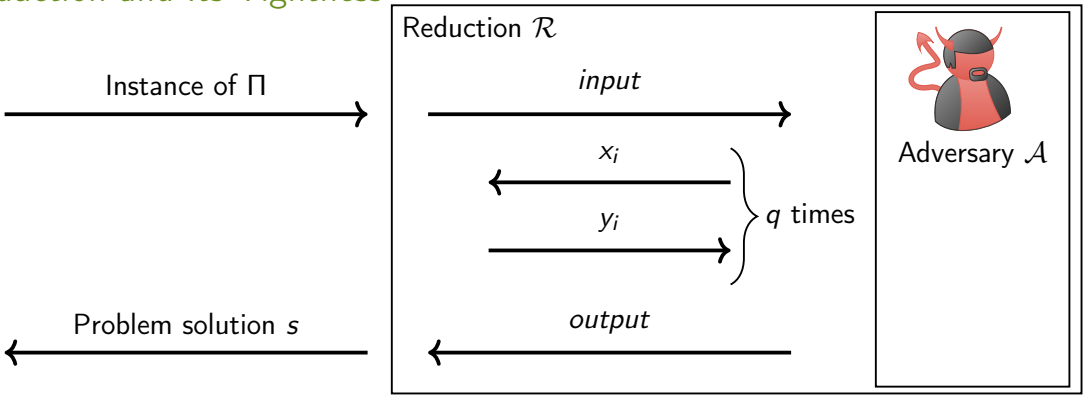
BERGISCHE
UNIVERSITÄT
WUPPERTAL

# Outline

# Reduction and Its Tightness

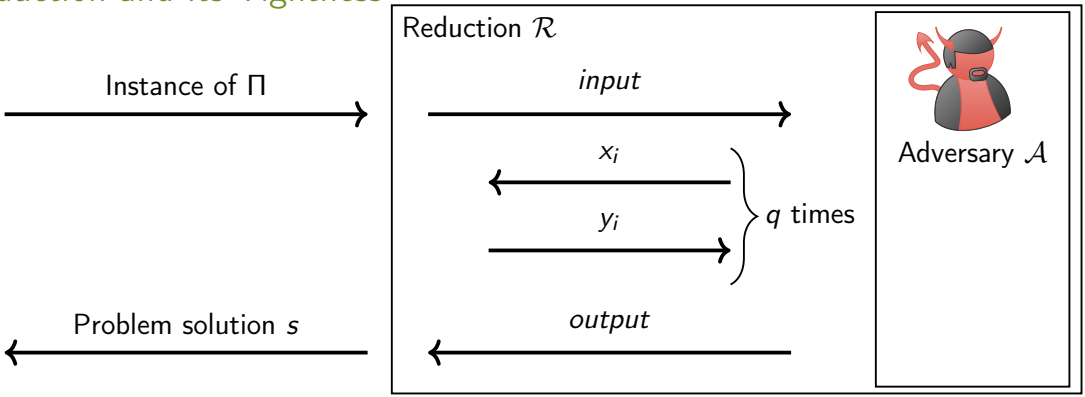


# Reduction and Its Tightness




Reduction is "tight", if $\mathbf{Time}(\mathcal{R}^{\mathcal{A}}) \approx \mathbf{Time}(\mathcal{A}) \quad \wedge \quad \mathbf{Adv}(\mathcal{R}^{\mathcal{A}}) \approx \mathbf{Adv}(\mathcal{A})$

# Reduction and Its Tightness

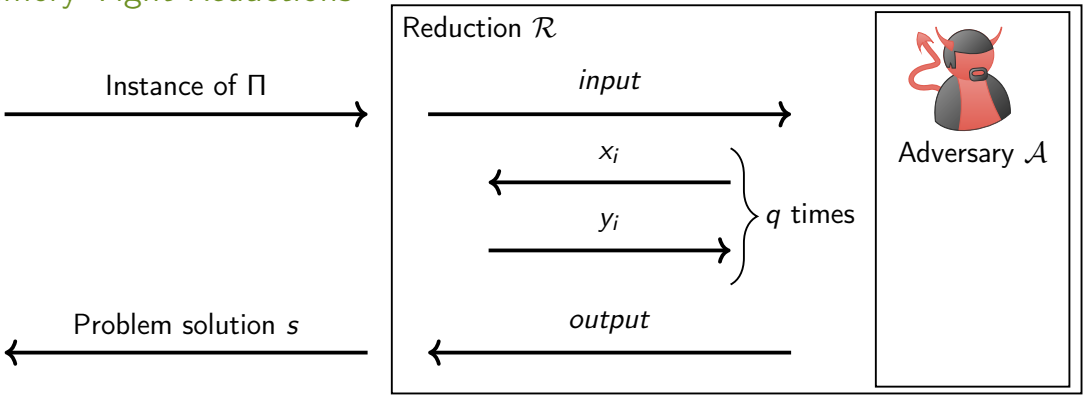



Reduction is "tight", if $\mathbf{Time}(\mathcal{R}^{\mathcal{A}}) \approx \mathbf{Time}(\mathcal{A}) \quad \wedge \quad \mathbf{Adv}(\mathcal{R}^{\mathcal{A}}) \approx \mathbf{Adv}(\mathcal{A})$

Often reductions are **not tight**, for example:

$\mathbf{Time}(\mathcal{R}^{\mathcal{A}}) \approx \mathbf{Time}(\mathcal{A})$ but $\mathbf{Adv}(\mathcal{R}^{\mathcal{A}}) \geq \mathbf{Adv}(\mathcal{A})/\ell$, the *security loss* $\ell$ depends on $\mathcal{A}$

# Memory-Tight Reductions

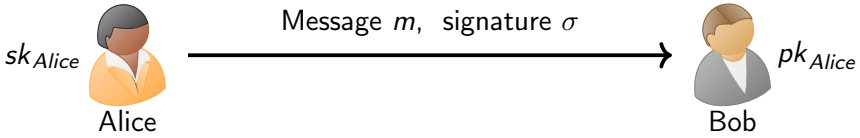



Reduction is "tight", if $\mathbf{Time}(\mathcal{R}^{\mathcal{A}}) \approx \mathbf{Time}(\mathcal{A}) \quad \wedge \quad \mathbf{Adv}(\mathcal{R}^{\mathcal{A}}) \approx \mathbf{Adv}(\mathcal{A})$

Auerbach, Cash, Fersch, Kiltz [ACFK17, CRYPTO]

Reduction is "memory-tight", if $\mathbf{Mem}(\mathcal{R}^{\mathcal{A}}) \approx \mathbf{Mem}(\mathcal{A})$

# Digital Signatures




Key generation: $(pk, sk) \xleftarrow{\$} \mathsf{Gen}(1^\lambda)$

Message signing: $\sigma \xleftarrow{\$} \mathsf{Sign}(sk, m)$

Signature verification: $\mathsf{Vfy}(pk, m, \sigma) \in \{0, 1\}$

## Correctness

For all $(pk, sk) \xleftarrow{\$} \mathsf{Gen}(1^\lambda)$ and all $m \in \{0,1\}^*$:

$$\Pr\left[\mathsf{Vfy}(pk, m, \mathsf{Sign}(sk, m)) = 1\right] = 1$$

# (s)EUF-CMA Security

Challenger

$(pk, sk) \xleftarrow{\$} \mathsf{Gen}(1^\lambda)$

$pk$

$m_i$

$\sigma_i \xleftarrow{\$} \mathsf{Sign}(sk, m_i)$

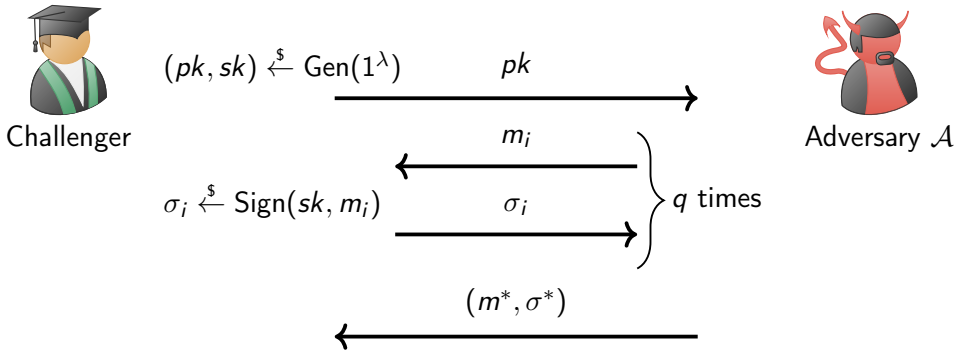
$\sigma_i$

$q$ times

$(m^*, \sigma^*)$

Adversary $\mathcal{A}$

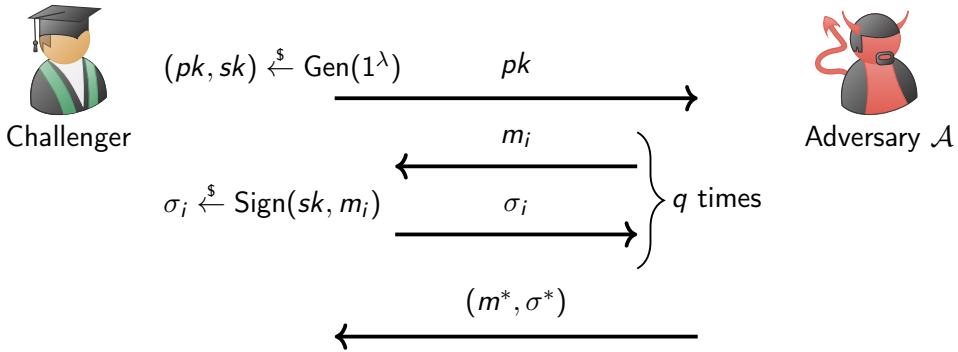
# (s)EUF-CMA Security




EUF-CMA: Adversary "wins", if:

$$\mathsf{Vfy}(pk, m^*, \sigma^*) = 1 \quad \wedge \quad m^* \notin \{m_1, \ldots, m_q\}$$

# (s)EUF-CMA Security



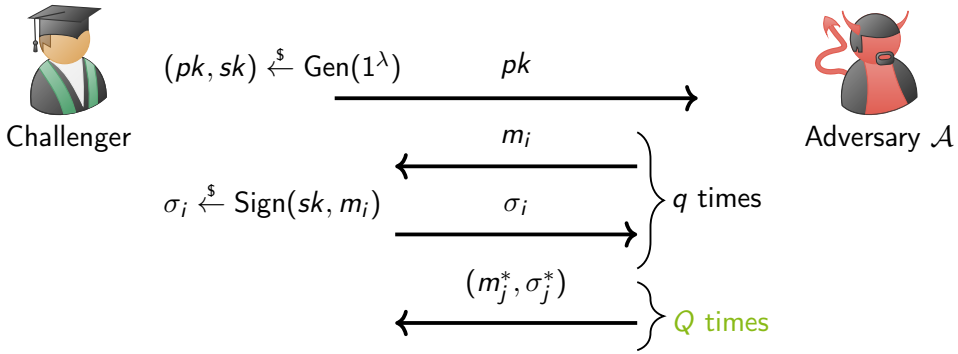

**EUF-CMA: Adversary "wins", if:**

$$\mathsf{Vfy}(pk, m^*, \sigma^*) = 1 \quad \wedge \quad m^* \notin \{m_1, \ldots, m_q\}$$

**sEUF-CMA: Adversary "wins", if:**

$$\mathsf{Vfy}(pk, m^*, \sigma^*) = 1 \quad \wedge \quad (m^*, \sigma^*) \notin \{(m_1, \sigma_1), \ldots, (m_q, \sigma_q)\}$$

# *Multi-Challenge* (s)EUF-CMA Security [Auerbach *et al.*, C'17]




**Adversary "wins", if:**

$$\exists j \text{ s.t. } \mathsf{Vfy}(pk, m_j^*, \sigma_j^*) = 1 \quad \wedge \quad \begin{cases} m_j^* \notin \{m_1, \ldots, m_q\} & \text{for mEUF-CMA} \\ (m_j^*, \sigma_j^*) \notin \{(m_1, \sigma_1), \ldots, (m_q, \sigma_q)\} & \text{for msEUF-CMA} \end{cases}$$

# *Multi-Challenge* (s)EUF-CMA Security [Auerbach *et al.*, C'17]

Challenger

Adversary $\mathcal{A}$

$(pk, sk) \xleftarrow{\$} \mathsf{Gen}(1^\lambda)$

$pk$ →

$m_i$ ←

$\sigma_i \xleftarrow{\$} \mathsf{Sign}(sk, m_i)$

$\sigma_i$ →

$q$ times

$(m_j^*, \sigma_j^*)$ ←

$Q$ times

**Adversary "wins", if:**

$$\exists j \text{ s.t. } \mathsf{Vfy}(pk, m_j^*, \sigma_j^*) = 1 \quad \wedge \quad \begin{cases} m_j^* \notin \{m_1, \ldots, m_q\} & \text{for mEUF-CMA} \\ (m_j^*, \sigma_j^*) \notin \{(m_1, \sigma_1), \ldots, (m_q, \sigma_q)\} & \text{for msEUF-CMA} \end{cases}$$

Memory-tightness is not obvious in this $Q$ times setting

# (Im)possibility of Signatures with Memory-Tight Multi-Challenge Security

Auerbach, Cash, Fersch, Kiltz [ACFK17, CRYPTO]:

- Certain natural black-box reductions from MC to *SC security* must be non-tight w.r.t memory or time
- Memory-tight reduction for RSA-FDH,
  but $\mathbf{Adv}(\mathcal{R}^{\mathcal{A}}) \geq q \cdot \mathbf{Adv}(\mathcal{A})$ and $\mathbf{Time}(\mathcal{R}^{\mathcal{A}}) \leq (q + q_H) \cdot \mathbf{Time}(\mathcal{A})$

# (Im)possibility of Signatures with Memory-Tight Multi-Challenge Security

Auerbach, Cash, Fersch, Kiltz [ACFK17, CRYPTO]:

- Certain natural black-box reductions from MC to *SC security* must be non-tight w.r.t memory or time
- Memory-tight reduction for RSA-FDH,
  but $\mathbf{Adv}(\mathcal{R}^{\mathcal{A}}) \geq q \cdot \mathbf{Adv}(\mathcal{A})$ and $\mathbf{Time}(\mathcal{R}^{\mathcal{A}}) \leq (q + q_H) \cdot \mathbf{Time}(\mathcal{A})$

Wang, Matsuda, Hanaoka, Tanada [WMHT18, EUROCRYPT]:

- Certain natural black-box reductions from MC security to *computational problems* must be non-tight w.r.t memory or time
- More lower bounds for MU-setting and collision-resistant hashing

# Our Approach

Auerbach *et al.* and Wang *et al.*:

*"Which properties of $\mathcal{R}$ are sufficient to **prove impossibility** of memory tightness?"*

Our work:

*"Which properties of $\mathcal{R}$ are sufficient to **achieve** memory tightness?"*

Which leverages do Auerbach *et al.* and Wang *et al.* leave us with?

- Consider non-black-box reductions
- Consider weaker (intermediate) security notions

# Our Approach

Auerbach *et al.* and Wang *et al.*:

*"Which properties of $\mathcal{R}$ are sufficient to **prove impossibility** of memory tightness?"*

Our work:

*"Which properties of $\mathcal{R}$ are sufficient to **achieve** memory tightness?"*

Which leverages do Auerbach *et al.* and Wang *et al.* leave us with?

- Consider non-black-box reductions
- Consider weaker (intermediate) security notions

**Main challenge:**

$\mathcal{R}$ must be able to distinguish "fresh" forgery from "replayed" message-signature pair, without using memory

# First Step: One-Signature-Per-Message (CMA1) Security

Challenger

$(pk, sk) \xleftarrow{\$} \mathsf{Gen}(1^\lambda)$

$pk$ →

← $m_i$

$\sigma_i \xleftarrow{\$} \mathsf{Sign}(sk, m_i)$

$\sigma_i$ →

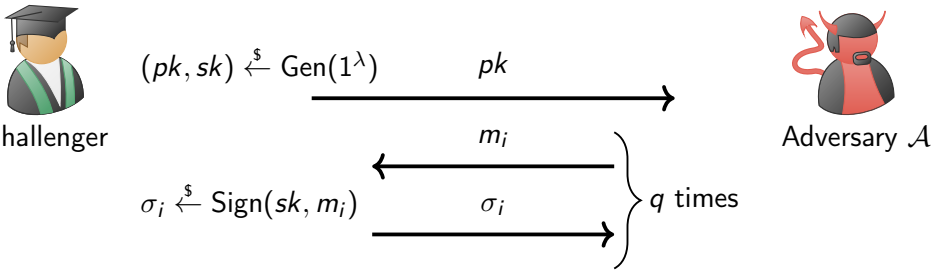
$q$ times

Adversary $\mathcal{A}$

**In the CMA1 Security Game:**

If $\exists i \neq i'$ s.t. $m_i = m_{i'}$, then $\sigma_i = \sigma_{i'}$

# First Step: One-Signature-Per-Message (CMA1) Security




**In the CMA1 Security Game:**

If $\exists i \neq i'$ s.t. $m_i = m_{i'}$, then $\sigma_i = \sigma_{i'}$

If there exists a reduction $\mathcal{R}$ for CMA1 security such that

- $\mathcal{R}$ simulates signature in some deterministic way

Then, intuitively,

- $\mathcal{R}$ can deal with $\mathsf{Sign}(m_i)$ and $\mathsf{Forge}(m^*, \sigma^*)$ queries without storing $\{(m_i, \sigma_i)\}$

# First Step: Canonical Reductions

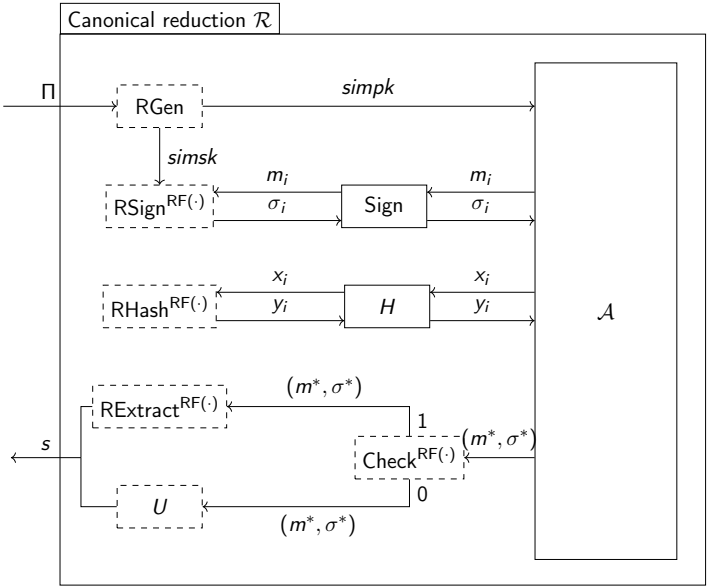
$\mathcal{R}$ **is** $(\ell, \delta)$**-canonical if:**

- It transfers any sEUF-CMA1 $\mathcal{A}$ into a hard problem solver

# First Step: Canonical Reductions

$\mathcal{R}$ **is** $(\ell, \delta)$**-canonical if:**

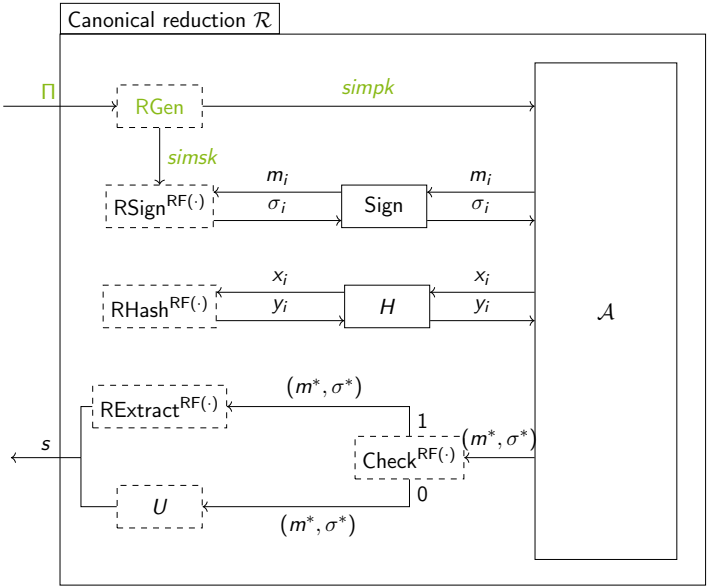
- It transfers any sEUF-CMA1 $\mathcal{A}$ into a hard problem solver
- It functions as shown in the right figure



# First Step: Canonical Reductions

$\mathcal{R}$ **is** $(\ell, \delta)$**-canonical if:**

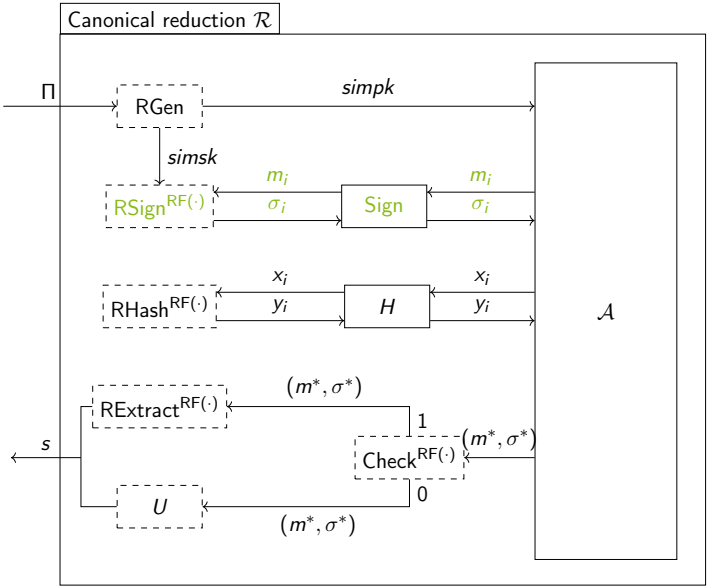
- It transfers any sEUF-CMA1 $\mathcal{A}$ into a hard problem solver
- It functions as shown in the right figure



# First Step: Canonical Reductions

$\mathcal{R}$ **is** $(\ell, \delta)$**-canonical if:**

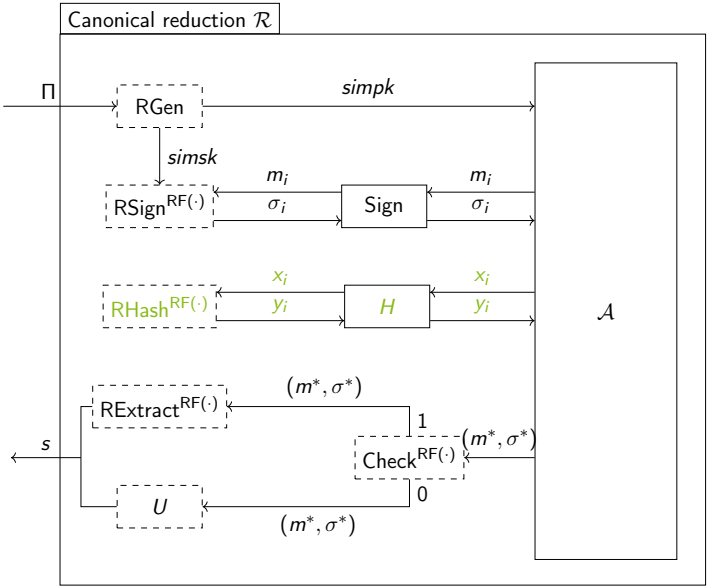
- It transfers any sEUF-CMA1 $\mathcal{A}$ into a hard problem solver
- It functions as shown in the right figure



# First Step: Canonical Reductions

$\mathcal{R}$ **is** $(\ell, \delta)$**-canonical if:**

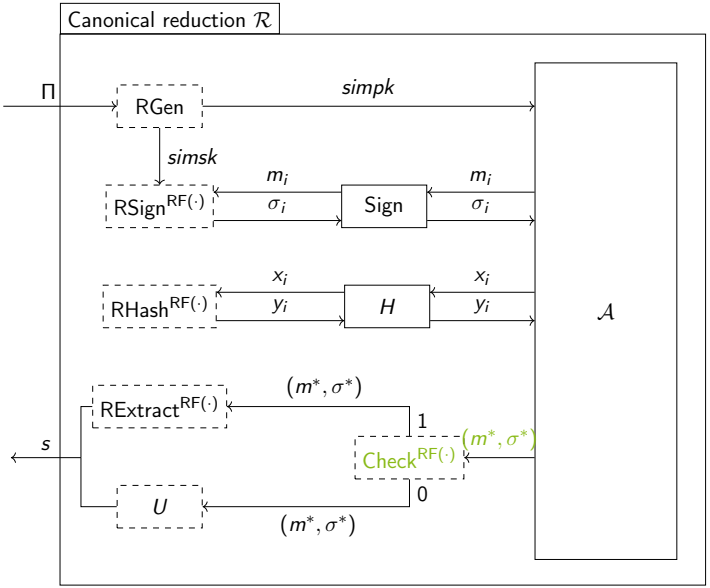
- It transfers any sEUF-CMA1 $\mathcal{A}$ into a hard problem solver
- It functions as shown in the right figure



# First Step: Canonical Reductions

$\mathcal{R}$ **is** $(\ell, \delta)$**-canonical if:**

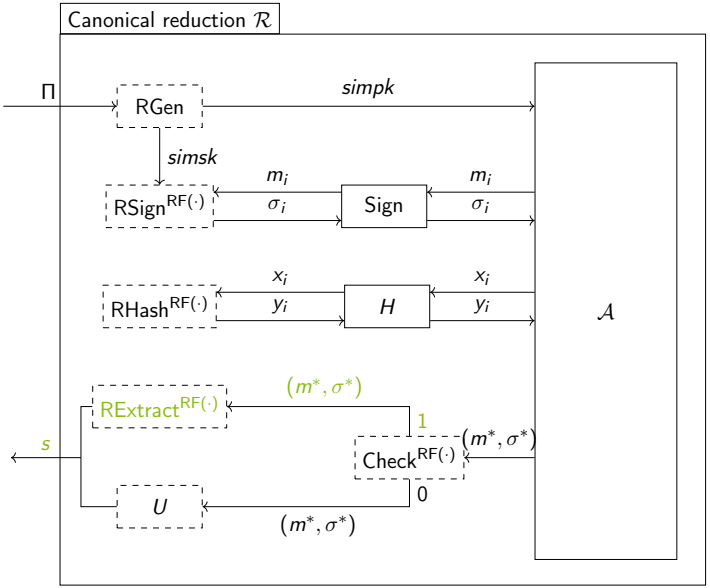
- It transfers any sEUF-CMA1 $\mathcal{A}$ into a hard problem solver
- It functions as shown in the right figure
- $\mathsf{Check}^{\mathsf{RF}(\cdot)}(m^*, \sigma^*) = 1$ iff $\begin{cases} \mathsf{Vfy}(\mathit{simpk}, m^*, \sigma^*) = 1 \wedge \\ \sigma^* \neq \mathsf{RSign}^{\mathsf{RF}(\cdot)}(\mathit{simsk}, m^*) \end{cases}$



# First Step: Canonical Reductions

$\mathcal{R}$ **is** $(\ell, \delta)$**-canonical if:**

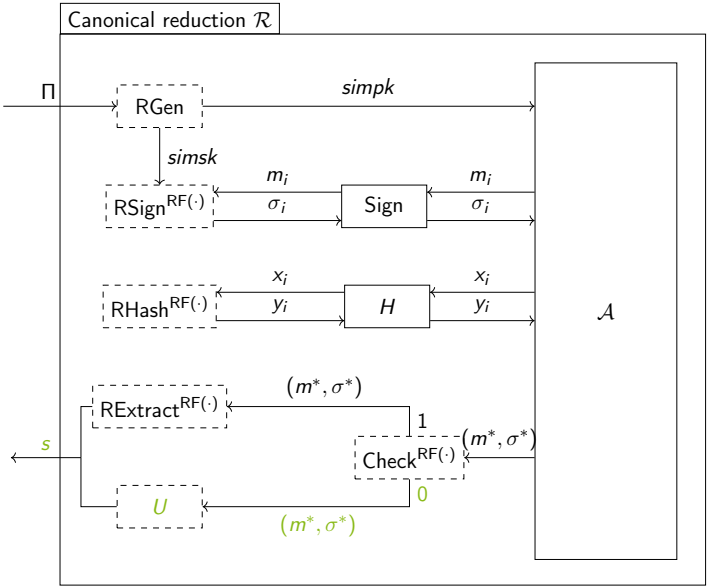
- It transfers any sEUF-CMA1 $\mathcal{A}$ into a hard problem solver
- It functions as shown in the right figure
- $\mathsf{Check}^{\mathsf{RF}(\cdot)}(m^*, \sigma^*) = 1$ iff $\begin{cases} \mathsf{Vfy}(\mathit{simpk}, m^*, \sigma^*) = 1 \wedge \\ \sigma^* \neq \mathsf{RSign}^{\mathsf{RF}(\cdot)}(\mathit{simsk}, m^*) \end{cases}$



# First Step: Canonical Reductions

$\mathcal{R}$ **is** $(\ell, \delta)$**-canonical if:**

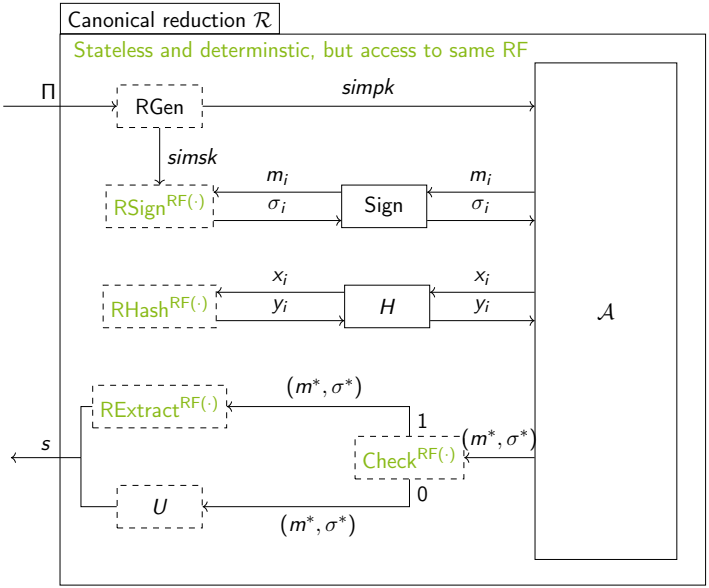
- It transfers any sEUF-CMA1 $\mathcal{A}$ into a hard problem solver
- It functions as shown in the right figure
- $\mathsf{Check}^{\mathsf{RF}(\cdot)}(m^*, \sigma^*) = 1$ iff $\begin{cases} \mathsf{Vfy}(\mathit{simpk}, m^*, \sigma^*) = 1 \wedge \\ \sigma^* \neq \mathsf{RSign}^{\mathsf{RF}(\cdot)}(\mathit{simsk}, m^*) \end{cases}$



# First Step: Canonical Reductions

$\mathcal{R}$ **is** $(\ell, \delta)$**-canonical if:**

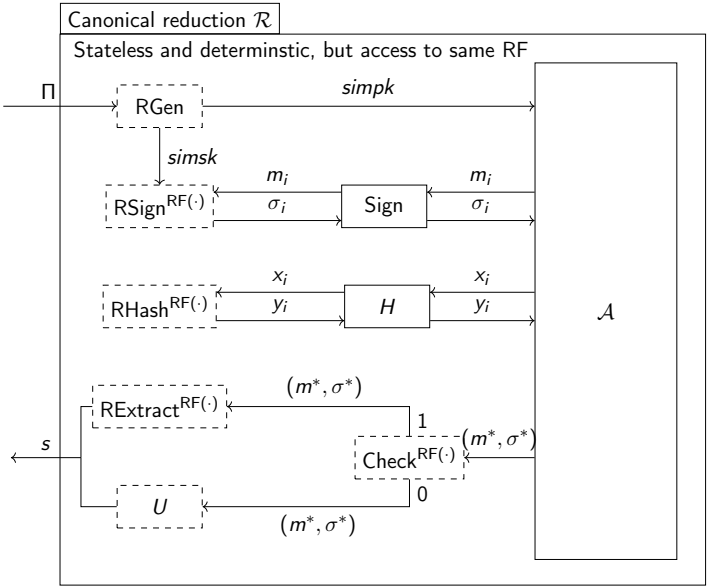
- It transfers any sEUF-CMA1 $\mathcal{A}$ into a hard problem solver
- It functions as shown in the right figure
- $\mathsf{Check}^{\mathsf{RF}(\cdot)}(m^*, \sigma^*) = 1$ iff $\begin{cases} \mathsf{Vfy}(\mathit{simpk}, m^*, \sigma^*) = 1 \wedge \\ \sigma^* \neq \mathsf{RSign}^{\mathsf{RF}(\cdot)}(\mathit{simsk}, m^*) \end{cases}$



# First Step: Canonical Reductions

$\mathcal{R}$ **is** $(\ell, \delta)$**-canonical if:**

- It transfers any sEUF-CMA1 $\mathcal{A}$ into a hard problem solver
- It functions as shown in the right figure
- $\mathsf{Check}^{\mathsf{RF}(\cdot)}(m^*, \sigma^*) = 1$ iff $\begin{cases} \mathsf{Vfy}(\mathit{simpk}, m^*, \sigma^*) = 1 \wedge \\ \sigma^* \neq \mathsf{RSign}^{\mathsf{RF}(\cdot)}(\mathit{simsk}, m^*) \end{cases}$
- $\mathbf{Adv}(\mathcal{R}^{\mathcal{A}}) \geq \mathbf{Adv}(\mathcal{A})/\ell - \delta$



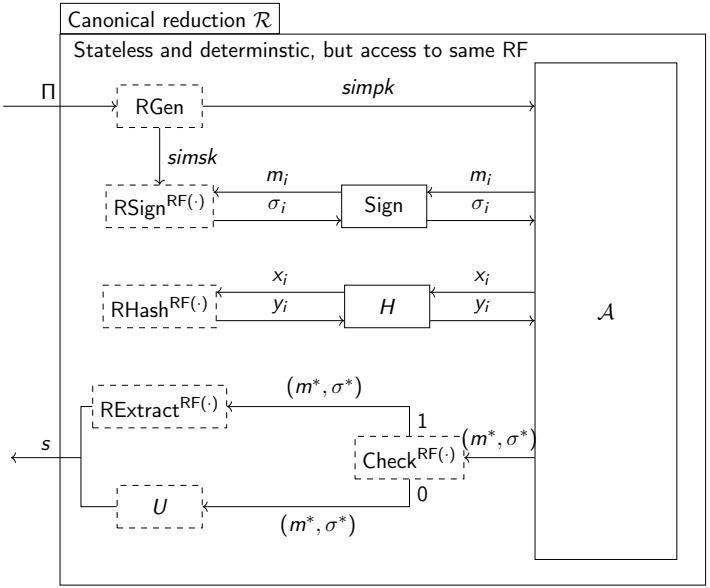
# First Step: Canonical Reductions

$\mathcal{R}$ **is** $(\ell, \delta)$**-canonical if:**

- It transfers any sEUF-CMA1 $\mathcal{A}$ into a hard problem solver
- It functions as shown in the right figure
- $\mathsf{Check}^{\mathsf{RF}(\cdot)}(m^*, \sigma^*) = 1$ iff $\begin{cases} \mathsf{Vfy}(\mathit{simpk}, m^*, \sigma^*) = 1 \wedge \\ \sigma^* \neq \mathsf{RSign}^{\mathsf{RF}(\cdot)}(\mathit{simsk}, m^*) \end{cases}$
- $\mathbf{Adv}(\mathcal{R}^{\mathcal{A}}) \geq \mathbf{Adv}(\mathcal{A})/\ell - \delta$

Canonical reductions are restricted

- single-challenge
- not memory-tight
- CMA1 instead of CMA



# First Step: Memory-Tight msEUF-CMA1 Reductions

## Main Theorem (Informal)

Let

- $\Sigma$ be a signature scheme,
- $\Pi$ be a *non-interactive problem*, and
- $\mathcal{R}$ be an $(\ell, \delta)$-*canonical reduction* from breaking the *single-challenge* sEUF-CMA1 security of $\Sigma$ to solving $\Pi$.

Using $\mathcal{R}$ and a memory-tightly-secure **pseudorandom function**, we can build a reduction $\mathcal{R}'$ from breaking the *multi-challenge* msEUF-CMA1 security of $\Sigma$ to solving $\Pi$, such that for any adversary $\mathcal{A}'$

$$\mathbf{Time}(\mathcal{R}'^{\mathcal{A}'}) \approx \mathbf{Time}(\mathcal{A}') \ \wedge \ \mathbf{Mem}(\mathcal{R}'^{\mathcal{A}'}) \approx \mathbf{Mem}(\mathcal{A}') \ \wedge \ \mathbf{Adv}(\mathcal{R}'^{\mathcal{A}'}) \approx \mathbf{Adv}(\mathcal{A}')/\ell - \delta$$

# First Step: Memory-Tight msEUF-CMA1 Reductions

## Main Theorem (Informal)

Let

- $\Sigma$ be a signature scheme,
- $\Pi$ be a *non-interactive problem*, and
- $\mathcal{R}$ be an $(\ell, \delta)$-*canonical reduction* from breaking the *single-challenge* sEUF-CMA1 security of $\Sigma$ to solving $\Pi$.

Using $\mathcal{R}$ and a memory-tightly-secure **pseudorandom function**, we can build a reduction $\mathcal{R}'$ from breaking the *multi-challenge* msEUF-CMA1 security of $\Sigma$ to solving $\Pi$, such that for any adversary $\mathcal{A}'$

$$\mathbf{Time}(\mathcal{R}'^{\mathcal{A}'}) \approx \mathbf{Time}(\mathcal{A}') \ \wedge \ \mathbf{Mem}(\mathcal{R}'^{\mathcal{A}'}) \approx \mathbf{Mem}(\mathcal{A}') \ \wedge \ \mathbf{Adv}(\mathcal{R}'^{\mathcal{A}'}) \approx \mathbf{Adv}(\mathcal{A}')/\ell - \delta$$

# First Step: Memory-Tight msEUF-CMA1 Reductions

## Main Theorem (Informal)

Let

- $\Sigma$ be a signature scheme,
- $\Pi$ be a *non-interactive problem*, and
- $\mathcal{R}$ be an $(\ell, \delta)$-*canonical reduction* from breaking the *single-challenge* sEUF-CMA1 security of $\Sigma$ to solving $\Pi$.

Using $\mathcal{R}$ and a memory-tightly-secure **pseudorandom function**, we can build a reduction $\mathcal{R}'$ from breaking the *multi-challenge* msEUF-CMA1 security of $\Sigma$ to solving $\Pi$, such that for any adversary $\mathcal{A}'$

$$\mathbf{Time}(\mathcal{R}'^{\mathcal{A}'}) \approx \mathbf{Time}(\mathcal{A}') \ \wedge \ \mathbf{Mem}(\mathcal{R}'^{\mathcal{A}'}) \approx \mathbf{Mem}(\mathcal{A}') \ \wedge \ \mathbf{Adv}(\mathcal{R}'^{\mathcal{A}'}) \approx \mathbf{Adv}(\mathcal{A}')/\ell - \delta$$

# First Step: Memory-Tight msEUF-CMA1 Reductions

## Main Theorem (Informal)

Let

- $\Sigma$ be a signature scheme,
- $\Pi$ be a *non-interactive problem*, and
- $\mathcal{R}$ be an $(\ell, \delta)$-*canonical reduction* from breaking the *single-challenge* sEUF-CMA1 security of $\Sigma$ to solving $\Pi$.

Using $\mathcal{R}$ and a memory-tightly-secure **pseudorandom function**, we can build a reduction $\mathcal{R}'$ from breaking the *multi-challenge* msEUF-CMA1 security of $\Sigma$ to solving $\Pi$, such that for any adversary $\mathcal{A}'$

$$\mathbf{Time}(\mathcal{R}'^{\mathcal{A}'}) \approx \mathbf{Time}(\mathcal{A}') \ \wedge \ \mathbf{Mem}(\mathcal{R}'^{\mathcal{A}'}) \approx \mathbf{Mem}(\mathcal{A}') \ \wedge \ \mathbf{Adv}(\mathcal{R}'^{\mathcal{A}'}) \approx \mathbf{Adv}(\mathcal{A}')/\ell - \delta$$

# First Step: Memory-Tight msEUF-CMA1 Reductions

## Main Theorem (Informal)

Let

- $\Sigma$ be a signature scheme,
- $\Pi$ be a *non-interactive problem*, and
- $\mathcal{R}$ be an $(\ell, \delta)$-*canonical reduction* from breaking the *single-challenge* sEUF-CMA1 security of $\Sigma$ to solving $\Pi$.

Using $\mathcal{R}$ and a memory-tightly-secure **pseudorandom function**, we can build a reduction $\mathcal{R}'$ from breaking the *multi-challenge* msEUF-CMA1 security of $\Sigma$ to solving $\Pi$, such that for any adversary $\mathcal{A}'$

$$\mathbf{Time}(\mathcal{R}'^{\mathcal{A}'}) \approx \mathbf{Time}(\mathcal{A}') \ \wedge \ \mathbf{Mem}(\mathcal{R}'^{\mathcal{A}'}) \approx \mathbf{Mem}(\mathcal{A}') \ \wedge \ \mathbf{Adv}(\mathcal{R}'^{\mathcal{A}'}) \approx \mathbf{Adv}(\mathcal{A}')/\ell - \delta$$

# First Step: Memory-Tight msEUF-CMA1 Reductions

## Main Theorem (Informal)

Let

- $\Sigma$ be a signature scheme,
- $\Pi$ be a *non-interactive problem*, and
- $\mathcal{R}$ be an $(\ell, \delta)$-*canonical reduction* from breaking the *single-challenge* sEUF-CMA1 security of $\Sigma$ to solving $\Pi$.

Using $\mathcal{R}$ and a memory-tightly-secure **pseudorandom function**, we can build a reduction $\mathcal{R}'$ from breaking the *multi-challenge* msEUF-CMA1 security of $\Sigma$ to solving $\Pi$, such that for any adversary $\mathcal{A}'$

$$\mathbf{Time}(\mathcal{R}'^{\mathcal{A}'}) \approx \mathbf{Time}(\mathcal{A}') \ \wedge \ \mathbf{Mem}(\mathcal{R}'^{\mathcal{A}'}) \approx \mathbf{Mem}(\mathcal{A}') \ \wedge \ \mathbf{Adv}(\mathcal{R}'^{\mathcal{A}'}) \approx \mathbf{Adv}(\mathcal{A}')/\ell - \delta$$

# First Step: Memory-Tight msEUF-CMA1 Reductions

## Main Theorem (Informal)

Let

- $\Sigma$ be a signature scheme,
- $\Pi$ be a *non-interactive problem*, and
- $\mathcal{R}$ be an $(\ell, \delta)$-*canonical reduction* from breaking the *single-challenge* sEUF-CMA1 security of $\Sigma$ to solving $\Pi$.

Using $\mathcal{R}$ and a memory-tightly-secure **pseudorandom function**, we can build a reduction $\mathcal{R}'$ from breaking the *multi-challenge* msEUF-CMA1 security of $\Sigma$ to solving $\Pi$, such that for any adversary $\mathcal{A}'$

$$\mathbf{Time}(\mathcal{R}'^{\mathcal{A}'}) \approx \mathbf{Time}(\mathcal{A}') \ \wedge \ \mathbf{Mem}(\mathcal{R}'^{\mathcal{A}'}) \approx \mathbf{Mem}(\mathcal{A}') \ \wedge \ \mathbf{Adv}(\mathcal{R}'^{\mathcal{A}'}) \approx \mathbf{Adv}(\mathcal{A}')/\ell - \delta$$

**Proof idea:** Construct $\mathcal{R}'$ from $\mathcal{R}$ by instantiating $\mathsf{RF}(\cdot)$ with the PRF and running "Check" for every Forge query made by $\mathcal{A}'$.

# First Step: Memory-Tight msEUF-CMA1 Reductions

## Main Theorem (Informal)

Let

- $\Sigma$ be a signature scheme,
- $\Pi$ be a *non-interactive problem*, and
- $\mathcal{R}$ be an $(\ell, \delta)$-*canonical reduction* from breaking the *single-challenge* sEUF-CMA1 security of $\Sigma$ to solving $\Pi$.

Using $\mathcal{R}$ and a memory-tightly-secure **pseudorandom function**, we can build a reduction $\mathcal{R}'$ from breaking the *multi-challenge* msEUF-CMA1 security of $\Sigma$ to solving $\Pi$, such that for any adversary $\mathcal{A}'$

$$\mathbf{Time}(\mathcal{R}'^{\mathcal{A}'}) \approx \mathbf{Time}(\mathcal{A}') \ \wedge \ \mathbf{Mem}(\mathcal{R}'^{\mathcal{A}'}) \approx \mathbf{Mem}(\mathcal{A}') \ \wedge \ \mathbf{Adv}(\mathcal{R}'^{\mathcal{A}'}) \approx \mathbf{Adv}(\mathcal{A}')/\ell - \delta$$

**Proof idea:** Construct $\mathcal{R}'$ from $\mathcal{R}$ by instantiating $\mathsf{RF}(\cdot)$ with the PRF and running "Check" for every Forge query made by $\mathcal{A}'$.

# Second Step: From msEUF-CMA1 to msEUF-CMA

Let $\Sigma' = (\mathsf{Gen}', \mathsf{Sign}', \mathsf{Vfy}')$ be a signature scheme. Define $\Sigma = (\mathsf{Gen}, \mathsf{Sign}, \mathsf{Vfy})$ as:

$\underline{\mathsf{Gen}(1^\lambda):}$
**Return** $\mathsf{Gen}'(1^\lambda)$

$\underline{\mathsf{Sign}(sk, m):}$
$n \xleftarrow{\$} \{0,1\}^{2\lambda}$
$\sigma' \xleftarrow{\$} \mathsf{Sign}'(sk, m \,\|\, n)$
**Return** $\sigma := (\sigma', n)$

$\underline{\mathsf{Vfy}(pk, m, \sigma):}$
$\mathsf{Parse}\ (\sigma', n) := \sigma$
**Return** $\mathsf{Vfy}'(pk, m \,\|\, n, \sigma')$

# Second Step: From msEUF-CMA1 to msEUF-CMA

Let $\Sigma' = (\mathsf{Gen}', \mathsf{Sign}', \mathsf{Vfy}')$ be a signature scheme. Define $\Sigma = (\mathsf{Gen}, \mathsf{Sign}, \mathsf{Vfy})$ as:

$\underline{\mathsf{Gen}(1^\lambda):}$
**Return** $\mathsf{Gen}'(1^\lambda)$

$\underline{\mathsf{Sign}(sk, m):}$
$n \xleftarrow{\$} \{0,1\}^{2\lambda}$
$\sigma' \xleftarrow{\$} \mathsf{Sign}'(sk, m \,\|\, n)$
**Return** $\sigma := (\sigma', n)$

$\underline{\mathsf{Vfy}(pk, m, \sigma):}$
$\mathsf{Parse}\ (\sigma', n) := \sigma$
**Return** $\mathsf{Vfy}'(pk, m \,\|\, n, \sigma')$

**Theorem (Informal)**

*If* $\Sigma'$ *is memory-tightly* msEUF-CMA1*-secure*

$\Longrightarrow$

$\Sigma$ *is memory-tightly* msEUF-CMA*-secure.*

# Second Step: From msEUF-CMA1 to msEUF-CMA

Let $\Sigma' = (\mathsf{Gen}', \mathsf{Sign}', \mathsf{Vfy}')$ be a signature scheme. Define $\Sigma = (\mathsf{Gen}, \mathsf{Sign}, \mathsf{Vfy})$ as:

$\underline{\mathsf{Gen}(1^\lambda):}$
**Return** $\mathsf{Gen}'(1^\lambda)$

$\underline{\mathsf{Sign}(sk, m):}$
$n \xleftarrow{\$} \{0,1\}^{2\lambda}$
$\sigma' \xleftarrow{\$} \mathsf{Sign}'(sk, m \,\|\, n)$
**Return** $\sigma := (\sigma', n)$

$\underline{\mathsf{Vfy}(pk, m, \sigma):}$
Parse $(\sigma', n) := \sigma$
**Return** $\mathsf{Vfy}'(pk, m \,\|\, n, \sigma')$

**Theorem (Informal)**

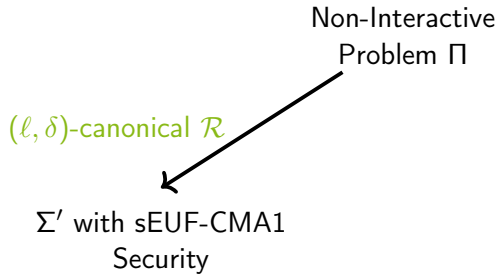
*If* $\Sigma'$ *is memory-tightly* msEUF-CMA1*-secure*

$\Longrightarrow$

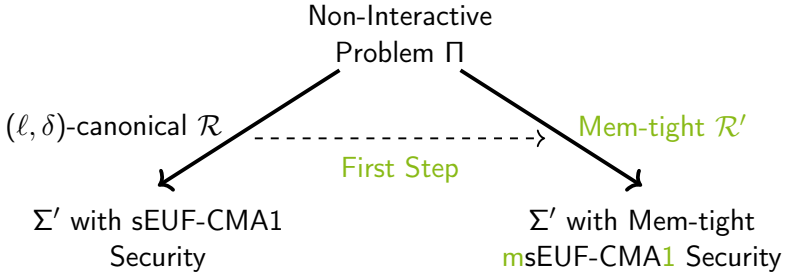
$\Sigma$ *is memory-tightly* msEUF-CMA*-secure.*

**Proof idea:** Nonces are uniformly random strings and are unlikely to repeat even when the same message is queried multiple times.

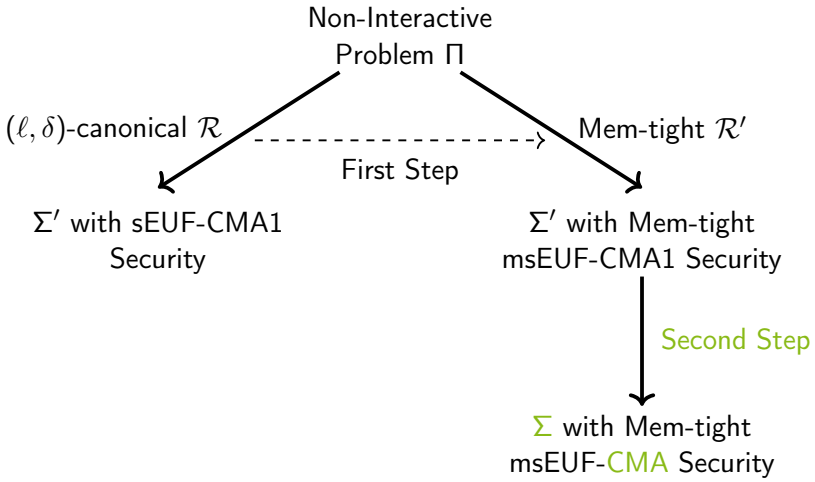
# Summary of Our Generic Approach



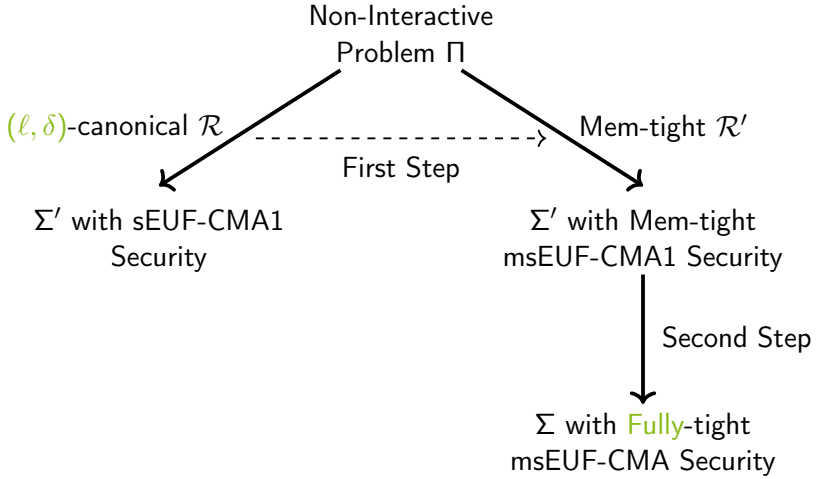
# Summary of Our Generic Approach



# Summary of Our Generic Approach



# Summary of Our Generic Approach



# Instantiations

We instantiate our approach and provide three *fully-tight* msEUF-CMA signature schemes

- The first scheme is based on the *lossy identification scheme* (LID) by Abdalla *et al.* [AFLT12, EUROCRYPT]
  - We show a $(1, \delta)$-canonical reduction for the LID-based signature scheme in [AFLT12]
  - Distinguishing lossy keys from normal keys is underlying non-interactive problem
  - $\delta = O(2^{-\lambda})$

# Instantiations

We instantiate our approach and provide three *fully-tight* msEUF-CMA signature schemes

- The first scheme is based on the *lossy identification scheme* (LID) by Abdalla *et al.* [AFLT12, EUROCRYPT]
  - We show a $(1, \delta)$-canonical reduction for the LID-based signature scheme in [AFLT12]
  - Distinguishing lossy keys from normal keys is underlying non-interactive problem
  - $\delta = O(2^{-\lambda})$
- The second scheme is the RSA-FDH scheme by Bellare and Rogaway [BR93, CCS]
  - The reduction Auerbach *et al.* provide in [ACFK17] for the RSA-FDH scheme can be seen as a $(\mathrm{e} \cdot q_S, 0)$-canonical reduction to the RSA problem
  - Katz and Wang in [KW03, CCS] propose a slight variant of RSA-FDH, which we call RSA-FDH+
  - We provide a $(2, 0)$-canonical reduction to the RSA problem for the RSA-FDH+ scheme

# Instantiations

We instantiate our approach and provide three *fully-tight* msEUF-CMA signature schemes

- The first scheme is based on the *lossy identification scheme* (LID) by Abdalla *et al.* [AFLT12, EUROCRYPT]
  - We show a $(1, \delta)$-canonical reduction for the LID-based signature scheme in [AFLT12]
  - Distinguishing lossy keys from normal keys is underlying non-interactive problem
  - $\delta = O(2^{-\lambda})$
- The second scheme is the RSA-FDH scheme by Bellare and Rogaway [BR93, CCS]
  - The reduction Auerbach *et al.* provide in [ACFK17] for the RSA-FDH scheme can be seen as a $(\mathrm{e} \cdot q_S, 0)$-canonical reduction to the RSA problem
  - Katz and Wang in [KW03, CCS] propose a slight variant of RSA-FDH, which we call RSA-FDH+
  - We provide a $(2, 0)$-canonical reduction to the RSA problem for the RSA-FDH+ scheme
- Similar results hold for the Boneh-Lynn-Shacham (BLS) scheme [BLS01, ASIACRYPT]

# Comparison

| Constr. | Proof | Asm. | Sec. | Sec. Loss | Mem. Loss | $\lvert\mathsf{pk}\rvert$ | $\lvert\sigma\rvert$ |
|---|---|---|---|---|---|---|---|
| LID-based | AFLT12 | DDH | EUF-CMA | $\mathcal{O}(1)$ | $\mathcal{O}(q_\mathsf{H}+q_\mathsf{S})$ | $4\lvert\mathbb{G}\rvert$ | $3\lvert\mathbb{Z}_q\rvert$ |
| | Ours | DDH | msEUF-CMA | $\mathcal{O}(1)$ | $\mathcal{O}(1)$ | $4\lvert\mathbb{G}\rvert$ | $3\lvert\mathbb{Z}_q\rvert+2\lambda$ |
| RSA-FDH | Coron00 | RSA | EUF-CMA | $\mathrm{e}\cdot q_\mathsf{S}$ | $\mathcal{O}(q_\mathsf{H}+q_\mathsf{S})$ | $\lvert N\rvert+\lvert e\rvert$ | $\lvert\mathbb{Z}_N\rvert$ |
| | ACFK17 | RSA | EUF-CMA | $\mathrm{e}\cdot q_\mathsf{S}$ | $\mathcal{O}(1)$ | $\lvert N\rvert+\lvert e\rvert$ | $\lvert\mathbb{Z}_N\rvert$ |
| | Ours | RSA | msEUF-CMA | $\mathrm{e}\cdot q_\mathsf{S}$ | $\mathcal{O}(1)$ | $\lvert N\rvert+\lvert e\rvert$ | $\lvert\mathbb{Z}_N\rvert$ |
| RSA-FDH+ | KatWan03 | RSA | EUF-CMA | $\mathcal{O}(1)$ | $\mathcal{O}(q_\mathsf{H}+q_\mathsf{S})$ | $\lvert N\rvert+\lvert e\rvert$ | $\lvert\mathbb{Z}_N\rvert$ |
| | Ours | RSA | msEUF-CMA | $\mathcal{O}(1)$ | $\mathcal{O}(1)$ | $\lvert N\rvert+\lvert e\rvert$ | $\lvert\mathbb{Z}_N\rvert+2\lambda$ |
| BLS | BonLynSha01 | (co-)CDH | EUF-CMA | $\mathrm{e}\cdot(q_\mathsf{S}+1)$ | $\mathcal{O}(q_\mathsf{H}+q_\mathsf{S})$ | $\lvert\mathbb{G}_2\rvert$ | $\lvert\mathbb{G}_1\rvert$ |
| | Ours | (co-)CDH | msEUF-CMA | $\mathrm{e}\cdot(q_\mathsf{S}+1)$ | $\mathcal{O}(1)$ | $\lvert\mathbb{G}_2\rvert$ | $\lvert\mathbb{G}_1\rvert$ |
| BLS+ | KatWan03 | (co-)CDH | EUF-CMA | $\mathcal{O}(1)$ | $\mathcal{O}(q_\mathsf{H}+q_\mathsf{S})$ | $\lvert\mathbb{G}_2\rvert$ | $\lvert\mathbb{G}_1\rvert$ |
| | Ours | (co-)CDH | msEUF-CMA | $\mathcal{O}(1)$ | $\mathcal{O}(1)$ | $\lvert\mathbb{G}_2\rvert$ | $\lvert\mathbb{G}_1\rvert+2\lambda$ |

# Concurrent Work and Acknowledgments

An independent and concurrent work

*Hiding in Plain Sight: Memory-tight Proofs via Randomness Programming*

by Ghoshal, Ghosal, Jaeger and Tessaro [GGJT21] studies the problem of getting memory-tight mEUF-CMA secure signature scheme via black-box reductions. Their construction is similar to ours in the second step, but their approach is completely different from ours.

# Concurrent Work and Acknowledgments

An independent and concurrent work

*Hiding in Plain Sight: Memory-tight Proofs via Randomness Programming*

by Ghoshal, Ghosal, Jaeger and Tessaro [GGJT21] studies the problem of getting memory-tight mEUF-CMA secure signature scheme via black-box reductions. Their construction is similar to ours in the second step, but their approach is completely different from ours.

We want to send our acknowledgments to

- Ghoshal, Ghosal, Jaeger and Tessaro for spotting a gap in the proof of our main theorem.
  - We close this gap in the full version of our paper by introducing a new property for canonical reduction and with a refined analysis.

# Concurrent Work and Acknowledgments

An independent and concurrent work

*Hiding in Plain Sight: Memory-tight Proofs via Randomness Programming*

by Ghoshal, Ghosal, Jaeger and Tessaro [GGJT21] studies the problem of getting memory-tight mEUF-CMA secure signature scheme via black-box reductions. Their construction is similar to ours in the second step, but their approach is completely different from ours.

We want to send our acknowledgments to

- Ghoshal, Ghosal, Jaeger and Tessaro for spotting a gap in the proof of our main theorem.
  - We close this gap in the full version of our paper by introducing a new property for canonical reduction and with a refined analysis.
- the anonymous reviewers of ASIACRYPT 2021 for insightful and helpful comments.

# Conclusions and Open Problems

To summarize our work.

- We propose a generic approach in getting memory-tight msEUF-CMA signatures.
- We instantiate our approach and get three signature schemes with fully-tight msEUF-CMA security.
- Our results do not conflict with the impossibility results by Auerbach *et al.* [ACFK17] or Wang *et al.* [WMHT18].

# Conclusions and Open Problems

To summarize our work.

- We propose a generic approach in getting memory-tight msEUF-CMA signatures.
- We instantiate our approach and get three signature schemes with fully-tight msEUF-CMA security.
- Our results do not conflict with the impossibility results by Auerbach *et al.* [ACFK17] or Wang *et al.* [WMHT18].

An interesting open problem

> *How to get a fully-tight or memory-tight multi-challenge secure signature in the standard model?*

# Conclusions and Open Problems

To summarize our work.

- We propose a generic approach in getting memory-tight msEUF-CMA signatures.
- We instantiate our approach and get three signature schemes with fully-tight msEUF-CMA security.
- Our results do not conflict with the impossibility results by Auerbach *et al.* [ACFK17] or Wang *et al.* [WMHT18].

An interesting open problem

> *How to get a fully-tight or memory-tight multi-challenge secure signature in the standard model?*

Full version: `https://eprint.iacr.org/2021/1220`

# Conclusions and Open Problems

To summarize our work.

- We propose a generic approach in getting memory-tight msEUF-CMA signatures.
- We instantiate our approach and get three signature schemes with fully-tight msEUF-CMA security.
- Our results do not conflict with the impossibility results by Auerbach *et al.* [ACFK17] or Wang *et al.* [WMHT18].

An interesting open problem

> *How to get a fully-tight or memory-tight multi-challenge secure signature in the standard model?*

Full version: `https://eprint.iacr.org/2021/1220`

Contact: {denis.diemert, kai.gellert, tibor.jager, lin.lyu}@uni-wuppertal.de

# References I


[ACFK17] Benedikt Auerbach, David Cash, Manuel Fersch, and Eike Kiltz, *Memory-tight reductions*, CRYPTO (1), Lecture Notes in Computer Science, vol. 10401, Springer, 2017, pp. 101–132.

[AFLT12] Michel Abdalla, Pierre-Alain Fouque, Vadim Lyubashevsky, and Mehdi Tibouchi, *Tightly-secure signatures from lossy identification schemes*, EUROCRYPT, Lecture Notes in Computer Science, vol. 7237, Springer, 2012, pp. 572–590.

[BLS01] Dan Boneh, Ben Lynn, and Hovav Shacham, *Short signatures from the weil pairing*, ASIACRYPT, Lecture Notes in Computer Science, vol. 2248, Springer, 2001, pp. 514–532.

[BR93] Mihir Bellare and Phillip Rogaway, *Random oracles are practical: A paradigm for designing efficient protocols*, CCS, ACM, 1993, pp. 62–73.

# References II


[GGJT21] Ashrujit Ghoshal, Riddhi Ghosal, Joseph Jaeger, and Stefano Tessaro, *Hiding in plain sight: Memory-tight proofs via randomness programming*, Cryptology ePrint Archive, Report 2021/1409, 2021, `https://ia.cr/2021/1409`.

[KW03] Jonathan Katz and Nan Wang, *Efficiency improvements for signature schemes with tight security reductions*, CCS, ACM, 2003, pp. 155–164.

[WMHT18] Yuyu Wang, Takahiro Matsuda, Goichiro Hanaoka, and Keisuke Tanaka, *Memory lower bounds of reductions revisited*, EUROCRYPT (1), Lecture Notes in Computer Science, vol. 10820, Springer, 2018, pp. 61–90.